Ejercicios de estimadores del núcleo de la función de densidad

Asignatura	Métodos Avanzados en Estadística
Correo	gloria.valle@estudiante.uam.es
Día	@November 15, 2021
Estudiante	Gloria del Valle Cano
Tema	Tema 2

Ejercicio 6

Sea $X_1, ..., X_n \;\; v.a.i.i.d$ de una distribución con densidad f. Se considera el estimador del núcleo $\hat{f}$ con núcleo rectangular $\mathcal{K}(x) = \mathbb{I}_{[-1/2,1/2]}(x)$ y parámetro de suavizado h.

(a) Calcula el sesgo y varianza de $\hat{f}$, para un valor de x fijo.

Teniendo que el estimador se puede expresar como:

$$\hat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x - X_i}{h}\right),$$

donde $h > 0$ y $\mathcal{K} \geq 0, \int \mathcal{K} = 1$.

Y que el kernel $\mathcal{K}$ se puede expresar también como:

$$\mathcal{K}\left(\frac{x-y}{h}\right) = \begin{cases} 1 \quad si \quad -\frac{1}{2} \leq \frac{x-y}{h} \leq \frac{1}{2} \\ 0 \qquad\qquad otherwise. \end{cases}$$

Lo que es lo mismo que decir que:

$$\mathcal{K}\left(\frac{x-y}{h}\right) = \begin{cases} 1 \quad si \quad -\frac{h}{2} + x \leq y \leq \frac{h}{2} + x \\ 0 \qquad\qquad otherwise. \end{cases} \qquad (*)$$

Calculamos por tanto el error (bias) y la varianza del estimador, partiendo de la propia definición:

- $Bias[\hat{f}(x)] = \mathbb{E}[\hat{f}(x)] - f(x) \qquad (1)$

$$\begin{aligned} \mathbb{E}[\hat{f}(x)] - f(x) &= \mathbb{E}\left[\frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x-X_i}{h}\right)\right] - f(x) \\ &= \frac{n}{nh}\,\mathbb{E}\left[\mathcal{K}\left(\frac{x-X_i}{h}\right)\right] \;-\; f(x) \\ &= \frac{1}{h}\int \mathcal{K}\left(\frac{x-y}{h}\right)f(y)dy \;-\; f(x) \\ (*) &= \int_{-\frac{h}{2}+x}^{\frac{h}{2}+x} f(y)dy - f(x) \\ &= \frac{1}{h}\left(F\left(x+\frac{h}{2}\right) - F\left(x-\frac{h}{2}\right)\right) - f(x) \end{aligned}$$

- $Var[\hat{f}(x)] = \mathbb{E}[\hat{f}(x) - \mathbb{E}[\hat{f}(x)]]^2 \qquad (2)$

$$
\begin{aligned}
\mathbb{E}[\hat{f}(x)-\mathbb{E}[\hat{f}(x)]]^2 &= \mathbb{E}\left[\frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x-X_i}{h}\right)-\mathbb{E}\left(\frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x-X_i}{h}\right)\right)\right]^2 \\
(\text{por linealidad de } \mathbb{E}) &= \mathbb{E}\left(\left(\frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x-X_i}{h}\right)\right)^2\right)-\mathbb{E}\left(\frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x-X_i}{h}\right)\right)^2 \\
(\mathcal{K}^2=\mathcal{K}) &= \mathbb{E}\left(\frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x-X_i}{h}\right)\right)-\mathbb{E}\left(\frac{1}{nh}\sum_{i=1}^{n}\mathcal{K}\left(\frac{x-X_i}{h}\right)\right)^2 \\
(1) &= \frac{1}{nh^2}\left(F\left(x+\frac{h}{2}\right)-F\left(x-\frac{h}{2}\right)\right)\left(1-\left(F\left(x+\frac{h}{2}\right)-F\left(x-\frac{h}{2}\right)\right)\right)
\end{aligned}
$$

(b) Demuestra que tanto el sesgo como la varianza tienden a cero si $h \to 0$ y $nh \to \infty$.

Por propia definición de derivada, aplicando límites llegamos a lo siguiente para el caso del sesgo:

$$
\begin{aligned}
\lim_{h\to 0}(1) &= \lim_{h\to 0}\left(\frac{F\left(x+\frac{h}{2}\right)-F\left(x-\frac{h}{2}\right)}{h}-f(x)\right) \\
&= \lim_{h\to 0}\Big(f(x)-f(x)\Big)=0
\end{aligned}
$$

Análogamente se resuelve para el caso de la varianza:

$$
\begin{aligned}
\lim_{nh\to\infty}(2) &= \lim_{nh\to\infty}\left(\frac{\left(F\left(x+\frac{h}{2}\right)-F\left(x-\frac{h}{2}\right)\right)\left(1-\left(F\left(x+\frac{h}{2}\right)-F\left(x-\frac{h}{2}\right)\right)\right)}{nh^2}\right) \\
&= \lim_{nh\to\infty}\left(\frac{1}{nh}f(x)(1-f(x))\right)=0
\end{aligned}
$$

Ejercicio 7

Considera una variable aleatoria con distribución beta de parámetros $\alpha = 3, \beta = 6$.

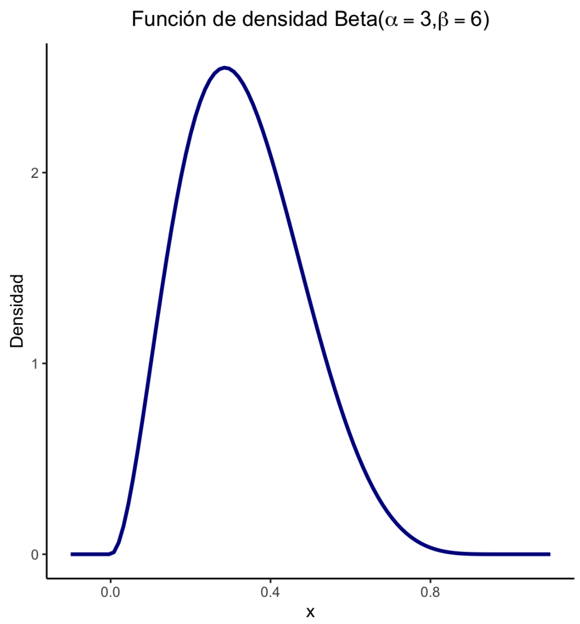
(a) Representa gráficamente la función de densidad y la función de distribución.

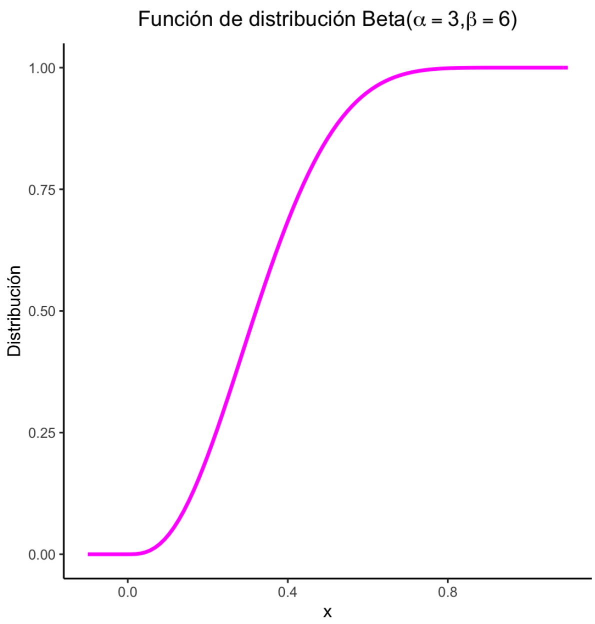
```
alpha <- 3
beta <- 6

# Función de densidad
graf1 <- ggplot()+
  ggtitle(TeX(r'(Función de densidad $Beta(\alpha = 3, \beta = 6)$)')) +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_function(fun = dbeta, args = list(alpha, beta), size = 1.1, col = 'darkblue') +
  labs(x = 'x', y = 'Densidad')+
  xlim(-0.1, 1.1)+
  theme(axis.line = element_line(colour = "black"),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank())

# Función de distribución
graf2 <- ggplot()+
  ggtitle(TeX(r'(Función de distribución $Beta(\alpha = 3, \beta = 6)$)')) +
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_function(fun = pbeta, args = list(alpha, beta), size = 1.1, col = 'magenta') +
  labs(x = 'x' , y = 'Distribución')+
  xlim(-0.1, 1.1)+
  theme(axis.line = element_line(colour = "black"),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank())

graf1+graf2
```





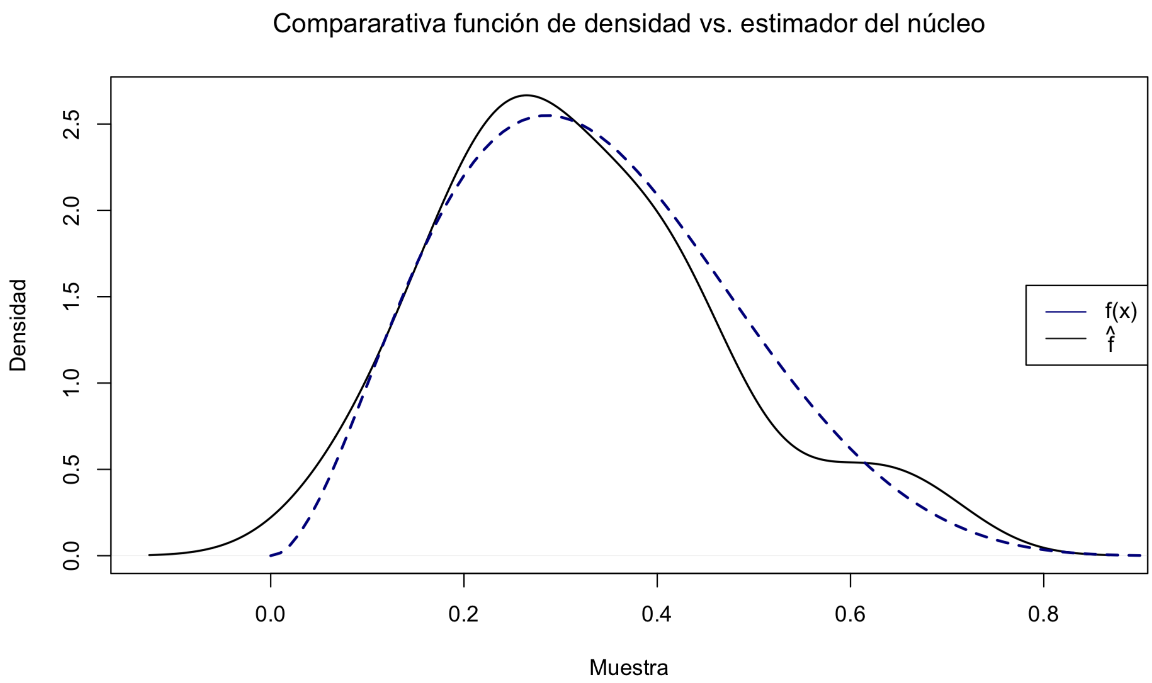
(b) Simula una muestra de tamaño 20 de esta distribución. A continuación representa en los mismos gráficos del apartado (a) las estimaciones de F y f obtenidas respectivamente mediante la función de distribución empírica F_n y un estimador del núcleo $\hat{f}$ obtenidos a partir de la muestra simulada.

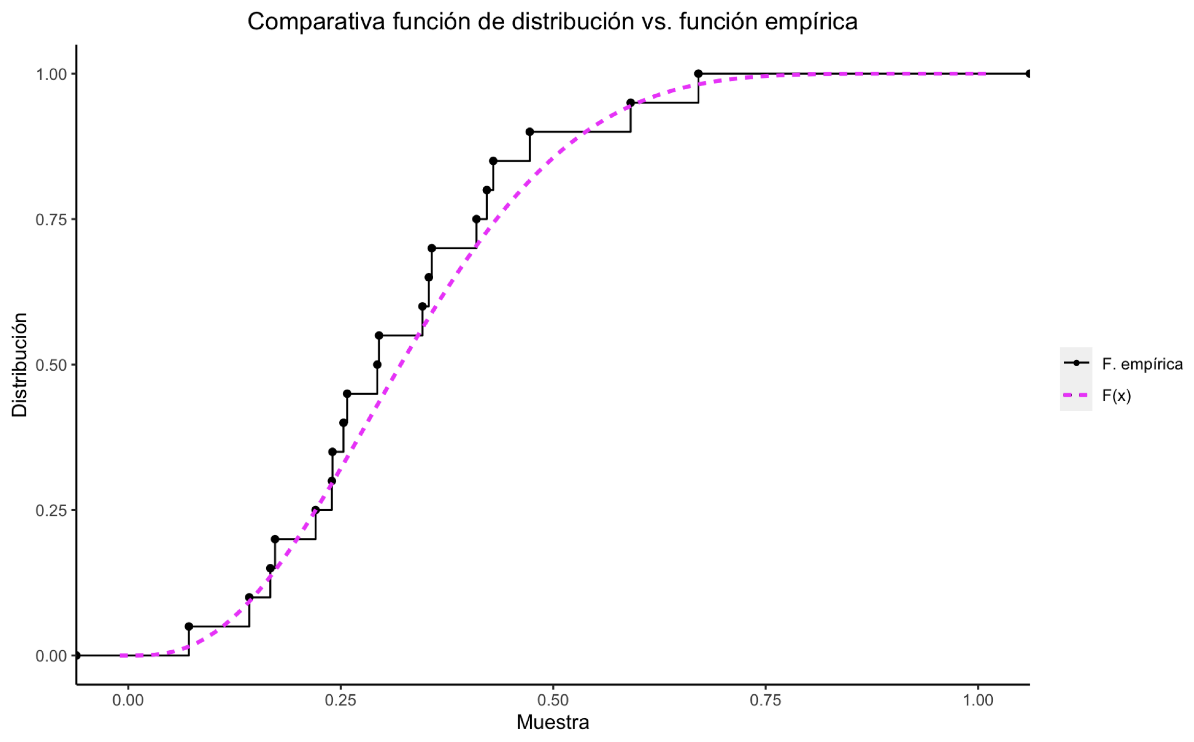
Se muestran sobre cada función la estimación correspondiente, observando que las estimaciones representan una buena aproximación sobre ambas PDF y CDF.

```
set.seed(123)
n <- 20
alpha <- 3
beta <- 6
muestra <- rbeta(n, alpha, beta)

# Estimación de pdf con estimador del núcleo
e_nucleo <- density(muestra)
plot(e_nucleo, lwd = 1.5, font.main = 1,
     main = "Compararativa función de densidad vs. estimador del núcleo",
     xlab = "Muestra", ylab = "Densidad")
curve(dbeta(x, alpha, beta), from = 0, to = 1, lty = 2,
              add = TRUE, col = 'darkblue', lwd = 2)
leg <- c('f(x)', TeX(r'($\hat{f}$)'))
legend("right", legend = leg, lty=c(1,1), col=c('darkblue','black'))

# Estimación de cdf con la función empírica
df <- data.frame(muestra)
ggplot(df, aes(muestra))+
  ggtitle('Comparativa función de distribución vs. función empírica') +
  theme(plot.title = element_text(hjust = 0.5)) +
  stat_ecdf(aes(muestra, linetype='F. empírica')) +
  stat_ecdf(aes(muestra), geom = 'point') +
  geom_function(fun = pbeta, args = list(alpha, beta), size = 1,
                col = 'magenta', aes(linetype = 'F(x)')) +
  labs(x = 'Muestra', y = 'Distribución', linetype = '') +
  xlim(-0.01, 1.01) +
  theme(axis.line = element_line(colour = "black"),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(),
        panel.border = element_blank(),
        panel.background = element_blank())
```





(c) Verifica empíricamente el grado de aproximación alcanzado en las estimaciones de F y f. Para ello, genera 200 muestras de tamaño 20 y para cada una de ellas evalúa el error (medido en la norma del supremo, es decir, el máximo de las diferencias entre las funciones) cometido al aproximar F por F_n y f por $\hat{f}$. Por último, calcula el promedio de los 200 errores obtenidos.

Evaluamos la precisión con el test Kolmogorov-Smirnov y así obtenemos los errores cometidos en ambos casos. Obtenemos tras ello el error máximo y el promedio para cada uno de los casos.

```
set.seed(123)
n <- 20
m <- 200
alpha <- 3
beta <- 6
```

```
error_F1 <- NULL
error_f2 <- NULL

for (i in 1:m){
  muestra <- rbeta(n, alpha, beta)

  ks_F <- ks.test(muestra, "pbeta", alpha, beta)
  error_F1 <- c(error_F1, ks_F$statistic)

  e <- dbeta(muestra, alpha, beta)
  nucleo <- density(muestra, n=20)$y

  ks_f <- ks.test(nucleo, e)
  error_f2 <- c(error_f2, ks_f$statistic)
}
```

```
> max(unlist(error_F1))
[1] 0.4049169
> max(unlist(error_f2))
[1] 0.65
> mean(error_F1)
[1] 0.1835117
> mean(error_f2)
[1] 0.50225
```